(12) UK Patent Application (19) GB (11) 2 329 489 (13) A

(43) Date of A Publication 24.03.1999

(21) Application No 9719829.5

(22) Date of Filing 17.09.1997

(71) Applicant(s)
Oxford Forecasting Services Limited
(Incorporated in the United Kingdom)
Saxon Beck, 59 St.Aldates, OXFORD, OX1 1ST, United Kingdom

(72) Inventor(s)
Benedict Seifert
Robert Hesselbo

(74) Agent and/or Address for Service
J A Kemp & Co.
14 South Square, Gray's Inn, LONDON, WC1R 5LX, United Kingdom

(51) INT CL[6]
G06F 17/60

(52) UK CL (Edition Q)
G4A AUXF

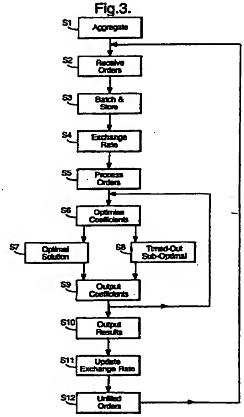
(56) Documents Cited
WO 96/34357 A1

(58) Field of Search
UK CL (Edition P) G4A AUXF
INT CL[6] G06F 17/60
Online: WPI, Banking Information Source, Computer

(54) Abstract Title
**Order processing apparatus and method**

(57) In an apparatus and method for processing trading orders, orders are received S2 for resources to be traded and stored S3 as an array; the orders are processed to calculate a set of coefficients each representing the proportion of each order which is to be satisfied; the values of the coefficients are optimized S6 with respect to at least one predetermined adjustable constraint and at least one predetermined adjustable criterion; the optimized coefficients stored and the processed orders output S10 with their respective coefficients. If a fully optimised set cannot be found within a specified period of time, a sub-optimal set of coefficients may be used. The criterion may be to maximise the volume of trade or the revenue of the broker. The constraint may be that the coefficients must be between 0 and 1 inclusive.

Fig.3.




GB 2 329 489 A

At least one drawing originally filed was informal and the print reproduced here is taken from a later filed formal copy.

This print incorporates corrections made under Section 117(1) of the Patents Act 1977.

Fig.1.

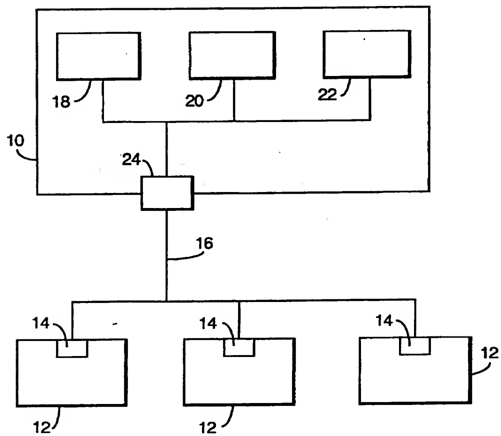


Fig.2.

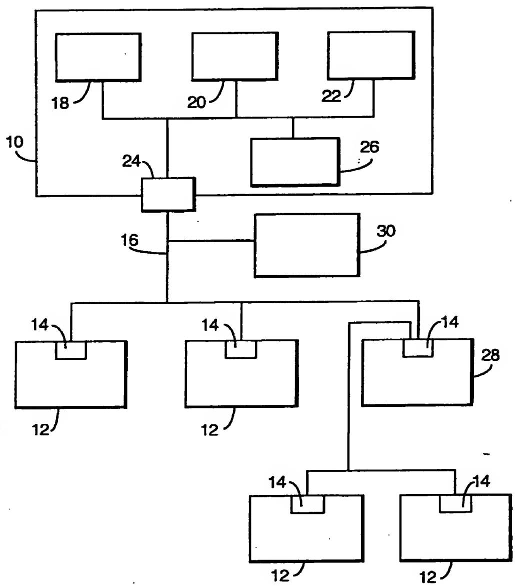
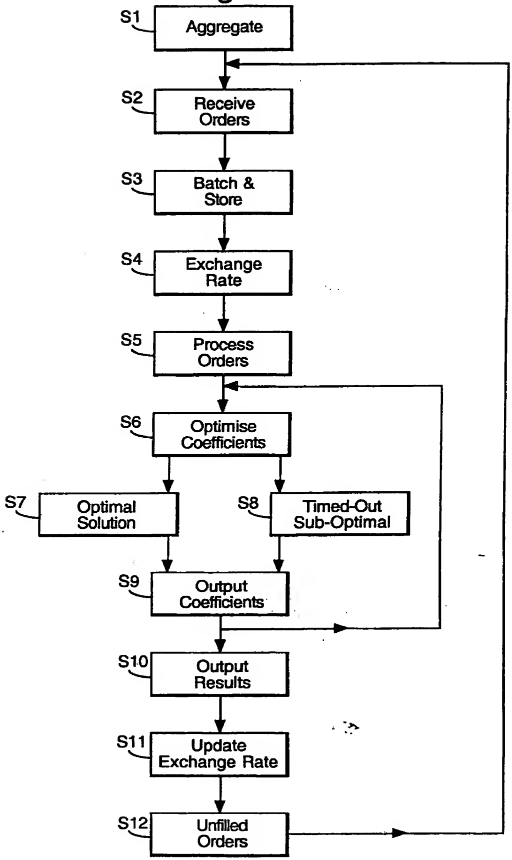


Fig.3.


S1
Aggregate
S2
Receive Orders
S3
Batch & Store
S4
Exchange Rate
S5
Process Orders
S6
Optimise Coefficients
S7
Optimal Solution
S8
Timed-Out Sub-Optimal
S9
Output Coefficients
S10
Output Results
S11
Update Exchange Rate
S12
Unfilled Orders

2329489



# ORDER PROCESSING APPARATUS AND METHOD

The present invention relates to an apparatus and method for optimising the allocation of resources based on received orders.

A wide variety of systems exists in which a set of users each submit orders which consist of a desired quantity of a given objective or resource which it is willing to exchange for a certain other quantity of another resource. Examples include a computer scheduling system allocating computing resources to users or to jobs submitted by users; electricity generating plants offering to supply power to a distribution system at different costs and generated from different fuels; a computer processor allocating resources such as memory and I/O bandwidth to different internal processes or software applications; and financial traders offering to buy and sell resources or financial instruments, such as stocks or currencies, in exchange for other financial instruments.

A number of different technical solutions have been used for the above allocation or matching problem. One example is binary matching in which an offer by one user to sell a particular quantity of one resource is matched to an offer by a different user to buy that quantity of that resource. A second example, in the case of scheduling computer jobs, is to allocate a slice of processor time to each user in turn on a rotational basis.

These solutions suffer from a number of drawbacks, namely that generally speaking the allocation of resources is not optimal. For example, supply and demand are not matched as well as they could be, such as in a computer, the majority of CPU cycles are idle, but at peak load the operation of the computer is limited by one of the resources available such as processor time, memory, I/O bandwidth. Scheduling systems do not necessarily take into account the priority of jobs, such as whether they are required in real time or can be batch processed, and allocating a time slice to each user in

rotation is simply a compromise. The matching of orders is inefficient in that in a binary matching system the size of the order must be matched so that a very large order may never be matched unless it is broken down into a number of smaller orders. The matching is also inefficient because in a binary matching system matches involving more than 2 instruments can not in general be found which prevents optimal matches from being discovered. In a financial market this can lead to illiquidity which in turn, and counter-intuitively, can lead to the problem of market volatility.

The present invention seeks to alleviate at least partially some or all of the above problems.

Accordingly the present invention provides an apparatus for processing trading orders, said apparatus comprising:

a central server;

a plurality of terminals on which user orders are to be entered; and

communication means for transmitting user orders between said terminals and said central server via a network;

wherein said central server comprises:

first storage means for storing received user orders as an array whose elements define the quantity of a particular first resource ordered by a particular user;

second storage means for storing an array of coefficients each representing the proportion of a particular order that is to be satisfied;

processing means for retrieving said orders from said first storage means, calculating an optimized set of values of said coefficients with respect to at least one predetermined, adjustable constraint and at least one predetermined, adjustable criterion, storing said optimized coefficient values in said second storage means; and

output means for communicating the processed orders and their respective coefficients.

According to a further aspect of the present invention

there is provided a method for processing trading orders comprising the steps of:

receiving from users orders each specifying the quantity of a particular first resource ordered by a particular user and storing them as an array in a first storage means;

processing said orders retrieved from said first storage means to calculate a set of coefficients each representing the proportion of a particular order that is to be satisfied;

optimizing the values of said coefficients with respect to at least one predetermined, adjustable constraint and at least one predetermined, adjustable criterion;

storing said optimized coefficient values in a second storage means; and

outputting the processed orders and their respective coefficients.

Embodiments of the present invention can produce optimised solutions to the matching of orders for resources. The solutions are not sensitive to the sizes of particular received orders. The matching can be done in an un-biased manner and is transparent to the particular resources being traded. Embodiments of the present invention that improve the efficiency of a trading system can alleviate the problem of illiquidity.

Embodiments of the invention will now be described by way of example only with reference to the accompanying drawings in which:

Figure 1 shows an apparatus according to the present invention;

Figure 2 shows a presently preferred embodiment of an apparatus according to the invention; and

Figure 3 is a flowchart illustrating a method according to the invention.

Referring to Figure 1, one form of apparatus according to the invention comprises a central server 10 and a plurality of terminals 12 on which user orders are to be entered. The terminals 12 can be conventional personal

computers (PC's) running appropriate software, or they may be dedicated trading terminals. Each terminal 12 is equipped with communication means 14 such as an interface and/or modem for transmitting the orders from the terminals 12 to the central server 10 via a network 16.

The central server 10 comprises a number of storage means. These may be devices such as memory chips or magnetic discs. The different storage means may comprise different regions within a common chip or disc or maybe distributed among several different physical devices. In particular the central server comprises a first storage means 18 for storing received user orders as an array whose elements define the quantity of a particular first resource ordered by a particular user and second storage means 20 for storing an array of coefficients each representing the proportion of a particular order that is to be satisfied.

The central server further comprises processing means 22, such as a central processing unit (CPU) executing instructions of a particular piece of software. The processing means 22 is for retrieving orders from the first storage means 18, calculating an optimised set of values of the coefficients with respect to at least one pre-determined, adjustable constraint and at least one pre-determined, adjustable criterion, and storing said optimised coefficient values in the second storage means 20.

An output means 24, such as an interface and/or modem, is comprised in the central server 10 for communicating the processed orders and their respective coefficients. Communications between the terminals 12 and central server 10 via the network 16 are preferably done via universal standards and protocols, such as TCP/IP and use a generally accepted interface, for instance an Internet browser. The communications may be done via a network, such as the internet, or an intranet to which all users of the broking system will be linked, and which is controlled by a central server. The orders entered by users are automatically passed

on to the server in real time, and are matched in batches (whose length and frequency is controlled in software by the server) in an optimal sense as explained in this paper.

A second, presently preferred embodiment of the invention is illustrated in Figure 2. Referring to Figure 2, this embodiment includes all the features shown in Figure 1, numbered with the same reference numerals, and several additional features. The additional features may be used with the apparatus of Figure 1 either individually or in combination. The central server 10 further comprises a third storage means 26 for storing an array of data representing the current exchange rate between each traded resource and at least one other resource which exchange rate data can be retrieved by said processing means 22. The processing means 22 may also compute and update the exchange rate data in the third storage means 26 based on the satisfied order flow as described later. The new orders stored in said storage means 18 further define a particular second resource offered in exchange for said first resource.

As shown in Figure 2, at least one of the terminals 12 is connected to the central server 10 via a sub-server 28 which aggregates orders from users before transmitting them to the central server 10. More than one sub-server 28 can be provided and they may be geographically separated such that each sub-server 28 aggregates orders from users in one particular region.

The output means 24 further communicates the processed orders and their respective coefficients to a further apparatus 30 for the settlement of the orders. The further apparatus 30 performs functions such as debiting and crediting the bank accounts of the users in accordance with the satisfied orders...

In the preferred embodiment, the system requires a broking house which participates in the order process by taking the opposite position from users in respect of the matched user orders. In this way the broking house will be

the formal counterparty for all trades executed in the system.

The matching algorithm makes sure that the broking house takes no market risk, and that subject to that condition, the matching is optimal with respect to certain criteria explained below.

The essential function of the optimisation algorithm which lies at the heart of the system is to simulate an "infinitely clever" broking house which is able to search through the order flow available to it in such a way as to satisfy the limit orders in the system to the optimal extent possible. Since there are several possible criteria which determine, from the view point of the broking house, what is optimal, the approach taken in this system is that of "cascading" a number of optimisation criteria. This is done in a way that can be controlled in software. In other words, the broking house (which runs the system) will be able to rank the optimisation criteria, from the most to the least important. Given this ranking, the algorithm will first search for the best overall matching within the meaning of the first optimisation criterion. Having found a set of optimal solutions, it will then search for a smaller set which are also optimal for the second criterion and so on. The most natural setting recommended by the system is to first optimise for volume (for the maximal portion of the order flow that can be satisfied) and then for the broker's revenue. It is assumed that the broker will earn fees simply by maximising the broker's revenue. However, other solutions are possible and can be chosen in real time by the operator or regulator.

The system is capable of being run in hierarchial form, where individual users may sign on via an intermediary institution, such as a clearing bank, which sets their credit limits, and determines margin accounts, communicates these credit limits to the central computer, and thereby causes orders to the truncated if they exceed credit limits.

The system is able to deal with both raw instruments i.e. instruments with no time component) and derivatives (instruments whose value at certain times in the future depends on the value of other "raw" instruments). The "no risk for the broker" criterion is defined in two different ways, depending on whether derivatives are involved or not involved in the application.

If all instruments are "raw", i.e. there are no derivatives involved, then the constraints take the form: the broker's aggregate trade in the batch involves no negative coefficients. In other words, the broker holds only non-negative amounts in each instrument separately.

If there are derivatives involved in the batch then the broker is declared to have a risk free-position if, under all scenarios of exercise or partial exercise of the options held by the other participants in the market, the broker can exercise his own derivatives in such a way that, at the end of the maturities involved in all derivatives held by other participants in the batch, the broker will hold a portfolio in the underlying instruments which has no negative coefficients. In other words, by playing his cards right, i.e. by making appropriate choices throughout the period during which derivatives involved in the batch may be exercised the broker can guarantee a risk-free portfolio in the underlying instrument space at the end.

The method according to an embodiment of the invention will now be described with reference to the flowchart of Figure 3.

At step S1 orders are received from users at a sub-server and are aggregated and transmitted to the central server. In step S2 the central server receives orders directly from users and aggregated by sub-servers in step S1 and unfilled orders from previous batches. The central server forms the received orders into a batch. The end of a batch is determined either by the volume of orders exceeding a threshold or a fixed time having elapsed since the previous batch. In step S3 the batch of orders are stored in a first

storage means.

In step S4 the processor of the central server retrieves exchange rate data from a third storage means and also retrieves the batch of orders from the first storage means. In step S5 the orders are processed to calculate a set of coefficients, each coefficient representing the proportion of a particular order that is to be satisfied.

The coefficients are then optimised at step S6. The optimisation is subject to constraints such as the coefficients must be less than or equal to 1 and greater than or equal to 0 and ensuring that the broker is not exposed to any risk. The coefficients are optimised with respect to a particular criterion, such as, maximising the total volume of satisfied orders and maximising the brokers revenue. The optimisation is performed by a software module. If an optimal solution is found the system proceeds to step S7. If a preset period of time elapses without an optimal solution being found the system proceeds to step S8.

In step S7 the optimising routine outputs data representing the optimised coefficients. In step S8 the optimising routine outputs data representing a sub-optimal set of coefficient values.

In step S9 the data representing the coefficient values output in step S7 or step S8 is stored in a second storage means. A number of different optimisation criteria are successively applied (usually 2, corresponding to maximising the volume and maximising the brokers revenue). If all the optimisation criteria have been applied the system proceeds to step S10, otherwise the system returns to step S6 to optimise the output coefficient data with respect to the next criterion.

In step S10 the resulting processed orders and their coefficients are output. This output is communicated to the users and to a mechanism for the settlement of the orders.

In step S11 the processor of the central server calculates new exchange rate data for the resources being

traded based on the satisfied order flow. The new exchange rate data is used to update that in the third storage means. The exchange rate data may also be communicated to the users of the system.

In step S12 orders that are unfilled, either completely or partially, and which have not been withdrawn by the user that submitted them and for which a specified period of time has not elapsed since they were submitted are returned to be processed in the next batch with any new orders received from users at step S2.

An example of the functioning of the apparatus and of the method according to the present invention will now be given. In this simple example, three different resources are being traded between four users (P,Q,R,S).

A particular batch of orders would be stored as the following array in the first storage means 18:

$$\begin{bmatrix} P & 1 & -1 & 0 \\ Q & 0 & 1 & -1 \\ R & -1 & 0 & 1 \\ S & -1.2 & 0 & 1 \end{bmatrix}$$

The first column contains the user code (P,Q,R,S). Each row represents an example of an order entered by a user on a terminal 12 and transmitted to the central server 10. The coefficients in columns 2 to 4 designate the amounts of the resources that the user wishes to acquire (positive sign) or give up (negative sign). In the example, user A wishes to acquire one unit of resource (I) (second column) (say a currency or a future on a stock index) for a maximum price of one unit of resource (II) (third column). The fundamental order matrix denoted F would be:

$$F=\begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ -1 & 0 & 1 \\ -1.2 & 0 & 1 \end{bmatrix}$$

A user has the option of ordering a desired quantity of a first resource at the prevailing market exchange rate (a quantity which is computed by a best fit method from previous batches by the system and delivered to the user's terminal in real time) in terms of a specific quantity of the second resource. On the other hand, the user has the option to ignore this prevailing market exchange rate and define his own exchange rate. In the illustrated example, users R and S order the same quantity of resource (III) for different amounts of resource (I), which illustrates the fact that user orders need not be related to the prevailing market exchange rate. If an order were made at the prevailing market exchange rate, the CPU 22 would obtain the exchange rate from the third storage means 26. The system will always compute the prevailing exchange rate in real time by a best-fit method using satisfied orders only, as explained later.

Having received the order flow and coded it as the matrix F above, the system will now proceed to produce an optimal matching. The matching produced by the system from the order matrix will be recorded in the second storage means 20, which contains an array MAT with 4 rows and 3 columns (in the general case N rows and k columns where N is the number of orders and k is the number of resources and k and N will always have this meaning):

$$MAT=\begin{bmatrix} 1 & 1 & 1 \\ 2 & 1 & 1 \\ 3 & 0 & 0 \\ 4 & 1 & 1 \end{bmatrix}$$

The coefficients in the second and third column will lie between 0 and 1. This constraint corresponds to the requirements that the user must not get more of a resource than she has ordered, and that she will never receive a negative amount of any resource ordered. The second constraint is that the broking house (which takes the opposite position from the market) must have non-negative amounts of each resource. This corresponds to the requirement that the broking house must bear no risk or exposure, i.e. no short positions in any instrument, without counting previous or outstanding positions the broking house may have. The entries in the third column are further constrained by the corresponding ones in the second column as discussed below. The first column merely represents order numbers and the system keeps track of which user submitted which order numbers.

The first column of MAT denotes an identification code for the number of the order in the batch. The second column denotes the degree to which the ordered quantity of the order can be matched (in this case orders 1, 2 and 4 will be completely matched and order 3 remains totally unmatched) and the third column denotes the degree to which the quantity of the resource offered in payment has been accepted in payment. As order 3 has not been supplied at all, of course order 3 gives rise to no payment, hence the third column third row coefficient is 0.

In general, there is a constraint that MAT[j,3] must be less than or equal to MAT[j,2] since a user can never be forced to pay at a higher exchange rate than that consented to by his order. When the system operates in "natural mode", MAT[j,3] equals MAT[j,2], i.e. the system always charges at precisely the exchange rate specified by the user. However, the system can operate in "spread control mode" in which inequality between the second and third columns is possible and it is by this mechanism that it can be guaranteed that the broker's revenue resulting from matching can not exceed

given thresholds.

The CPU 22 performs computations for optimising the coefficients of columns 2 and 3 of the matrix MAT defined above, which codes for the degree to which orders are to be accepted. This computation is done by means of a cascade of optimisation criteria controlled by different apparatuses denoted Ap(i). Each apparatus may be a software routine or module. The routines must be arranged sequentially, Ap(1),..., Ap(n). The first routine Ap(1) takes the order batch F and then returns the set of permissible N by 3 coefficient matrices, all of which satisfy the first optimisation criterion, and passes these onto routine Ap(2), and so on until Ap(n) outputs a particular coefficient matrix denoting an optimal coefficient matrix MAT.

In the default case, this cascade will be of length 2 (but the broker can modify the length). The most important optimisation criteria are liquidity (i.e. the total value of the executed trades in the matching) and broker's revenue, i.e. the spread or difference between what is paid in and what is paid out. This is how the broker will earn money within the system. A particular apparatus will now be described in detail with respect to the present example.

Ap(1) will take the order matrix and operate the following transformation on it. We shall denote the number of resources by k and the number of user orders in the batch by N.

First, the matrix D = [B;C] is produced, where B is 8 (in general, 2N) by 4 N dimensional, and C is simply the transpose of the matrix F above (the rows of C correspond to resource indices, the columns to user orders). In general the rows of B will consist of 1 times the j-th unit vector which has a 1 in column j and 0's elsewhere (for rows with index 2j-1), and -1 times the j-th unit vector (for rows with index 2j).

In the example these matrices are as follows:

$$B=\begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

and

$$C=\begin{bmatrix} 1 & 0 & -1 & -1.2 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 1 \end{bmatrix}$$

The 1-dimensional array b is then produced where b is defined as:

$$b=[1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0]$$

In general, the first 2N entries of b are, alternately 1's and 0's, and the remaining entries are all 0.

An exchange rate matrix E is defined and is stored in third storage means 26. In this example it is a 1-dimensional array of length k:

E = [1,u,v]

The entries of E denote the values of 1 unit of resources I, II and III (in general 1,...k) in terms of the operator's home resource (which is always resource with lable 1, and hence the first entry of E is always equal to 1. The current exchange rate data itself is stored in terms of a square array such that the (j,k) entry denotes the exchange rate between resources j and k.

There is further an apparatus for producing a 1-dimensional array of length N, whose i-th coefficient will be given by the amount of the first resource of the i-th order,

multiplied by the relevant coefficient of the exchange rate matrix E. We call that vector OPT. It is a vector of length N.

Ap(1) then is an apparatus performing one of the known algorithms (e.g. polynomial time) returning a family of solutions to the problem defined by the optimisation problem defined by volume as a target function and by the constraints defined by the two criteria: orders can be satisfied to a degree between 0 and 1, and the broker's return must be non-negative in all instruments.

Sources of known algorithms for solving this problem are:

N.K. Karmarkar. A new polynomial-time algorithm for linear programming. Combinatorica, 4:373-395.

C. Roos. T. Terlaky, J-Ph Val Theory and Algorithms for Linear Optimisation, An Interior Point Approach, J. Wiley, 1997.

B. Jansen C. Roos, T. Terlaky, J-Ph Vial, Primal-Dual Algorithms for linear programming based on the logarithmic barrier method J. of Optimisation, Theory and Applications, 83:1-26, 1994.

R. Sedgewick, Algorithms in C++, Addison Wesley, 1992.

W.H. Press et al.,Numerical Recipes in C, 2nd ed. Cambridge, 1992.

More explicitly, this is given by the following definition (in terms of this example) of the optimisation routine involved (the "first optimisation routine"):

First optimisation routine: Optimise the real-valued function of a N-dimensional vector argument x, defined by:

f(x) = inner product of array OPT with x

and the constraints are:

$D.x \leq \text{transpose}(b)$

Where D is the (11 by 4 matrix) [B;C], i.e. the matrix whose first 8 rows are those of the matrix B (as above) and whose remaining 3 rows are those of C (as above).

The optimisation apparatus will produce as an output the set of permissible orders according to the first optimisation routine. The form of this output will be a matrix of the form given in this example as MAT above, but with the 2nd column replaced by the transposed vector x expressed as a linear function of parameters, with constraints on these parameters, and the 3rd column being set equal to the 2nd.

$$\begin{bmatrix} 1 & 1 & 1 \\ 2 & 1 & 1 \\ 3 & t & t \\ 4 & 1-t & 1-t \end{bmatrix}$$

In the example this is the following set of paramaterised matrices which we shall denote by o(t):

$$o(t) = \begin{bmatrix} P & 1 & -1 & 0 \\ Q & 0 & 1 & -1 \\ R & -t & 0 & t \\ S & -1.2*(1-t) & 0 & 1-t \end{bmatrix}$$

Here the parameter t ranges from 0 to 1.

The second stage of the optimisation then uses a similar apparatus to the one described for optimisation step 1, in order to optimise the function G(t) where

G(t) = Broker's revenue for order o(t)

G(t) is computed as minus the sum of the terms in each column of the matrix o(t), each evaluated at the broker's exchange rate.

The second stage of the optimisation can be explicitly defined in terms of apparatuses analogously to the

construction involved in the first optimisation step as follows.

First, the second stage of the optimisation will involve an apparatus for finding the constrained optimum of the following linear function of 2*N variables, where the first N co-ordinates refer, as above, to the coefficients coding for the degree to which orders are being satisfied, and the second N co-ordinates to the degree to which the offered payment is being accepted. For the purposes of the mode described herein, these degrees are the same, but in the general mode they may not be.

The role of the array OPT above is now played by an array OPT2 of twice that length whose coefficients are arrived at as follows:

Firstly, the order matrix is split into two matrices, whose sum is the order matrix, defined uniquely by the property that the first matrix has only non-negative coefficients and the second only non-positive ones. Further, for each of these matrices, and for each row, the sum over all columns is taken, with each entry multiplied by the exchange rate E as above, so that, for both the positive and negative matrix, an array of length N is arrived at. Juxtaposing these two arrays gives the array OPT2. In our example, this array is as follows:

$$\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ -1 \\ -1 \\ -1 \\ -1.2 \end{bmatrix}$$

In the example under discussion, the associated optimisation function of 8 variables is then the sum of the

first 4 parameters - next 3 parameters - 1.2 times the fourth parameter (designating the degree to which the payments involved in the 4 user orders have been accepted). The coefficients in this particular functional form come from the positive and negative component of the matrix F in the obvious way. This is the function which the apparatus seeks to optimise, using the matrix F as it arises from the order flow.

Next the constraints of this optimisation function are coded for in an apparatus which contains the constraints that the jth and N+jth co-ordinates must be equal (in the general case of spread control this constraint will be relaxed), and that the 1st N constraints in terms of the 1st N paramaters are the same as before, plus the new constraint that OPT takes the optimal value on these 1st N paramaters as that which was the result of optimisation step 1.

Thus, one sees that the only data that the apparatus involved in the first optimisation step needs to record is the value of the function OPT given the constraints described.

In the example, the final answer to the cascaded optimisation routine is the matrix.

$$\begin{bmatrix} P & 1 & -1 & 0 \\ Q & 0 & 1 & -1 \\ R & 0 & 0 & 0 \\ S & -1.2 & 0 & 1 \end{bmatrix}$$

Having established through the optimisation stage 1 that all of P and all of Q can be satisfied, and that a mixture of R and S (given by a parameter-t-between 0 and 1) is possible as far as optimisation criterion 1 (the volume at chosen operator exchange rate) is concerned, the 2nd optimisation criterion picks out the solution under which the operator receives the highest price for 1 unit of resource I thus maximising his revenue.

Whenever orders are satisfied, they are fed into the exchange rate apparatus, which maintains a store of recent satisfied orders in order to produce a best fit to the exchange rates of the most recent trades. An exemplary best fit algorithm is as follows:

-Satisfied orders are added to the store. Each order defines an edge in the "order network" whose nodes correspond to resources

-Remove the oldest order which can be removed without causing the order network to become disconnected (there are standard algorithms to determine if a network is connected, see e.g. Sedgewick, chapter 29). Repeat until all such orders have been removed.

-Find the vector L[i] which minimizes the sum over the remaining satisfied orders of the square of the difference between the logarithm of the exchange rate of the order and the difference between the elements of L corresponding to the resources of each order, weighted by the value of the order (there are standard algorithms to perform such least squares optimisations, see e.g. Press et al., chapter 15).

The exchange rate matrix exp(L[i]-L[j])is then output. Unlike conventional systems, offers and bids which have not been satisfied are completely ignored as far as the exchange rate computation is concerned.

The server tracks incoming orders in real time. The new orders being received are added to the existing batch of unfilled orders. A set of batch criteria can be defined by the user, most importantly simply the total volume computed in terms of the broking house's home currency, of the orders comprising the batch. When the criterion is met then the server will compute an optimal matching, using at its core a routine for solving linear programming problems, and

communicating the result of the batching (i.e. the vector of partial satisfactions) to both the broking house and the users.

In order to be sure to get an optimal matching in real time, the program allows for a menu of linear programming routines to be used. The program tries each of them for a specified maximal period of time. If the optimisation routine terminates within that time then it returns its optimal value. The program then chooses the best solution (in lexicographic order with respect to the set of optimisation criteria adopted, i.e. starting with the main optimisation criterion, (which in the default setting is volume) and working its way down through all the optimisation criteria adopted. The best solution is then adopted, and the matching recorded and communicated to the participants.

Unlike conventional broking systems, which are based on the construct of direct exchanges between counterparties, Midas allows the user to input a wide range of different sizes of orders. Throughout this paper, the term Midas will be used to designate the present invention. This is due to the fact that, from a mathematical standpoint, the optimisation problem is well defined and indeed invariant under the operations of splitting up a very large order into many small ones, or aggregating any finite number of small orders into one big order. Furthermore, there are effective numerical methods of optimisation which can practically implement the solution to the optimisation problem. As the structure of the most natural optimisation problems arising (as those listed above) is linear, any of the many numerical schemes for solving linear optimisation problems will do the job. The present invention has the advantage that it enables trades which are genuinely non-binary: they are not exchanges between two parties, but are "n-ary exchanges" of k commodities among n traders. A matching can occur between 3 users, P, Q and R, even if P and Q, Q and R, and P and R, as isolated pairs of users would not be in a position to trade.

For instance, if P, Q, R held, respectively, 1 unit of three commodities p, q, r, (stocks, options or currencies), but wanted to buy (respectively) q, r, p, at exchange rates all equal to 1, then the system would automatically produce a matching by assigning q to P, r to Q, and p to R.

Similar examples can be constructed which illustrate the general capacity of the system to construct matchings which will take direct advantage of the full range of market participation, both as regards the full set of users and the full set of instruments are concerned. Suppose, for the sake of argument, that there are at any given time users within the system putting in orders, involving 500,000 financial instruments. It is possible to construct examples which show that a matching is possible, and would in fact be constructed by the algorithm, where there would be no satisfiable orders, that is, no trading possible among any 499,999 users trading 499,999 different financial instruments. In other words, the matching algorithm allows for global transactions among an ensemble of users in a large-dimensional instrument market which could not be carried out if any subset of users were to bargain among themselves.

The matching according to these embodiments is not based on bargaining between individuals but on a numerical procedure by which a match, i.e. an allocation of resources among traders, is implemented, and this numerical procedure is centrally computed, not arrived at by conscious and hence specifically informed process involving the individual traders. It is for this reason that it is unnecessary for individual users to be aware of the other orders in the system. Thus anonymity can be preserved, which is a key advantage to users who can place orders without fearing that their entry into the market will perturb the market in a direction opposite to their own interest.

CLAIMS

1. An apparatus for processing trading orders, said apparatus comprising:

a central server;

a plurality of terminals on which user orders are to be entered; and

communication means for transmitting user orders between said terminals and said central server via a network;

wherein said central server comprises:

first storage means for storing received user orders as an array whose elements define the quantity of a particular first resource ordered by a particular user;

second storage means for storing an array of coefficients each representing the proportion of a particular order that is to be satisfied;

processing means for retrieving said orders from said first storage means, calculating an optimized set of values of said coefficients with respect to at least one predetermined, adjustable constraint and at least one predetermined, adjustable criterion, storing said optimized coefficient values in said second storage means; and

output means for communicating the processed orders and their respective coefficients.

2. An apparatus according to claim 1, wherein said at least one constraint includes that the value of each of said coefficients is less than or equal to 1 and greater than or equal 0.

3. An apparatus according to claim 1 or 2, wherein, said processing means is adapted to process orders such that a designated user takes the opposite position to each other user order by agreeing to exchange a proportion of the crdered quantity of said first resource for a second resource, where said proportion corresponds to the optimized coefficient for that order.

4. An apparatus according to claim 3, wherein said at least one constraint includes that if all orders were to be completed, in proportion to their respective coefficients, the designated user's holdings arising from the processed orders would be only non-negative amounts of each resource, including after maturation of all options to trade resources in the future.

5. An apparatus according to claim 3 or 4, wherein said at least one criterion includes maximizing the revenue of said designated user, in terms of a particular resource, based on the differences in exchange rates at which resources are traded.

6. An apparatus according to any one of the preceding claims, wherein said central server further comprises third storage means for storing an array of data representing the current exchange rate between each resource and at least one other resource, and wherein said processing means is further for retrieving exchange rate data from said third storage means.

7. An apparatus according to any one of the preceding claims, wherein said at least one criterion includes maximizing the total value, in terms of a particular resource, of orders satisfied, partially and in full.

8. An apparatus according to any one of the preceding claims, wherein said processing means is adapted to optimize the values of said coefficients by successively applying respective criteria in a cascaded manner.

9. An apparatus according to claim 6, further comprising means to specify the sequence of said cascaded criteria.

10. An apparatus according to any one of the preceding claims, wherein said processing means is adapted to apply, in sequence, each one of a plurality of predefined linear programming routines to optimize said coefficients until one of the following events occurs:

a specified maximum period of time elapses;
an optimal solution is found.

11. An apparatus according to claim 10, wherein if a specified maximum period of time elapses before an optimal solution is found, a consistent sub-optimal solution is used as the optimized set of coefficient values.

12. An apparatus according to any one of the preceding claims, wherein said processing means is adapted to optimize said coefficients for batches of received orders.

13. An apparatus according to claim 12, wherein said processing means is adapted to determine the end of a batch by a preset interval of time having elapsed since the start of that batch.

14. An apparatus according to claim 12, wherein said processing means is adapted to determine the end of a batch by the total order value exceeding a threshold value.

15. An apparatus according to any one of claims 12 to 14, adapted to carry forward orders not satisfied, completely or partially, in one batch to the next batch.

16. An apparatus according to any one of claims 12 to 15, adapted to remove from said first storage means orders not satisfied, completely or partially, after a preset length of time from submission of those orders.

17. An apparatus according to claim 16, wherein said

preset length of time for each order is specified by the relevant user.

18. An apparatus according to any one of the preceding claims, wherein unsatisfied orders are removed from said first memory means at the request of a user.

19. An apparatus according to any one of the previous claims, wherein said user orders stored in said first storage means define a particular second resource offered in exchange for said first resource.

20. An apparatus according to claim 19, wherein the quantity of said second resource is specified.

21. An apparatus according to claim 19, wherein said second resource is offered for said first resource at the prevailing market exchange rate.

22. An apparatus according to any one of the preceding claims, wherein at least one of said terminals is connected to said central server via a sub-server which aggregates orders from users.

23. An apparatus according to any one of the preceding claims, wherein said communication means is adapted to transmit orders using TCP/IP.

24. An apparatus according to any one of the preceding claims, wherein said processing means computes and updates the exchange rates in said third storage means based on the satisfied order flow.

25. An apparatus according to any one of the preceding claims, wherein said resources being traded are financial, such as currencies, securities, and futures on the value of

commodities.

26. An apparatus according to any one of the preceding claims, wherein the output of said output means is communicated to a further apparatus for settlement of said orders.

27. A method for processing trading orders comprising the steps of:

receiving from users orders each specifying the quantity of a particular first resource ordered by a particular user and storing them as an array in a first storage means;

processing said orders retrieved from said first storage means to calculate a set of coefficients each representing the proportion of a particular order that is to be satisfied;

optimizing the values of said coefficients with respect to at least one predetermined, adjustable constraint and at least one predetermined, adjustable criterion;

storing said optimized coefficient values in a second storage means; and

outputting the processed orders and their respective coefficients.

28. A method according to claim 27, wherein said at least one constraint includes that the value of each of said coefficients is less than or equal to 1 and greater than or equal to 0.

29. A method according to claim 27 or 28, wherein a designated one of said users takes the opposite position to each other user order by agreeing to exchange a proportion of the ordered quantity of said first resource for a second resource, where said proportion corresponds to the optimized coefficient for that order.

30. A method according to claim 29, wherein said at

least one constraint includes that if all orders were to be completed, in proportion to their respective coefficients, the designated user's holdings arising from the processed orders would be only non-negative amounts of each resource, including after maturation of all options to trade resources in the future.

31. A method according to claim 29 or 30, wherein said optimizing step includes as one criterion maximizing the designated user's revenue, in terms of a particular resource, based on the differences in the exchange rates at which resources are traded.

32. A method according to any one of claims 27 to 31, wherein a third storage means is for storing an array of data representing the current exchange rate between each resource and at least one other resource, said method further comprising the step of retrieving exchange rate data from a third storage means for use in optimizing said coefficients.

33. A method according to any one of claims 27 to 32, wherein said optimizing step includes maximizing the total value, in terms of a particular resource, of orders satisfied, partially and in full, as one criterion.

34. A method according to any one of claims 27 to 33, wherein said optimizing step further comprises successively applying respective criteria in a cascaded manner to obtain optimized values of said coefficient.

35. A method according to claim 34, further comprising the step of specifying the sequence of said cascaded criteria.

36. A method according to any one of claims 27 to 35, wherein said optimizing step further comprises applying, in

sequence, each one of a plurality of predefined linear programming routines to optimize said coefficients until one of the following events occurs:

a specified maximum period of time lapses;

an optimal solution is found.

37. A method according to claim 36, wherein if a specified maximum period of time elapses before an optimal solution is found, a consistent sub-optimal solution is used as the optimized set of coefficient values.

38. A method according to any one of claims 27 to 37, wherein said processing step further comprises retrieving said orders from said second storage means in batches, and is followed by said optimizing step to obtain optimized coefficient values for said batch of orders.

39. A method according to claim 38, wherein the end of a batch is determined by a preset interval of time since the start of that batch.

40. A method according to claim 38, wherein the end of a batch is determined by the total order value exceeding a threshold value.

41. A method according to any one of claims 38 to 40, further comprising the step of forwarding orders in one batch that are not satisfied, completely or partially, following the optimizing step, to be processed in the next batch.

42. A method according to any one of claims 38 to 41, further comprising the step of removing orders from said second storage means that have not been satisfied, completely or partially, after a preset length of time from submission of those orders.

43. A method according to claim 42, wherein said preset length of time for each order is specified by the relevant user.

44. A method according to any one of claims 27 to 42, further comprising the step of deleting from said second memory means unsatisfied orders at the request of a user.

45. A method according to any one of claims 27 to 44, wherein said user orders stored in said first storage means define a particular second resource offered in exchange for said first resource.

46. A method according to claim 45, wherein the quantity of said second resource is specified.

47. A method according to claim 45, wherein said second resource is offered for said ordered first resource at the prevailing market exchange rate.

48. A method according to any one of claims 27 to 47, further comprising the step of communicating orders entered on a plurality of terminals to a central server for processing said orders, via a network.

49. A method according to claim 48, further comprising the steps of aggregating in a sub-server orders from users before communicating them to said central server.

50. A method according to claim 48 or 49, wherein said communication is done by means of TCP/IP.

51. A method according to any one of claims 27 to 50, further comprising the step of computing updated exchange rates based on the satisfied order flow and storing said updated exchange rates in said third storage means.

52. A method according to any one of claims 27 to 51, wherein said resources being traded are financial, such as currencies, securities, and futures on the value of commodities.

53. A method according to any one of claims 27 to 52, further comprising the step of transmitting the result of said outputting step to a means for settlement of said orders.

54. An apparatus constructed and arranged to operate substantially as described herein with reference to the accompanying drawings.

55. A method substantially as described herein with reference to the accompanying drawings.

The Patent Office

50

| | | | |
|---|---|---|---|
| **Application No:** | GB 9719829.5 | **Examiner:** | Michael Richardson |
| **Claims searched:** | 1-55 | **Date of search:** | 24 February 1998 |

## Patents Act 1977
## Search Report under Section 17

**Databases searched:**

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.P): G4A (AUXF)

Int Cl (Ed.6): G06F 17/60

Other: Online: WPI, Banking Information Source, Computer

**Documents considered to be relevant:**

| Category | Identity of document and relevant passage | Relevant to claims |
|---|---|---|
| X | WO 96/34357 A1 (MIT HOLDINGS) See especially pages 14-15, 23-24 and 27-35 | 1, 2, 6, 8-10, 12, 15-21, 24-28, 32, 34-36, 38, 41-47, 51-53 at least |

| | | | |
|---|---|---|---|
| X | Document indicating lack of novelty or inventive step | A | Document indicating technological background and/or state of the art. |
| Y | Document indicating lack of inventive step if combined with one or more other documents of same category. | P | Document published on or after the declared priority date but before the filing date of this invention. |
| & | Member of the same patent family | E | Patent document published on or after, but with priority date earlier than, the filing date of this application. |